// 1 . Write a for loop that prints the even numbers from 1 to 20.

public class unit\_2\_1 {

    public static void main(String[] args) {

        for (int i = 1; i <= 20; i++) {

            if (i % 2 == 0) {

                System.out.println(i);

            }

        }

    }

}

/\*  2 .Create a while loop that prompts the user for their flight choice until a valid number is

entered\*/

import java.util.Scanner;

public class unit\_2\_2 {

    public static void main(String[] args) {

        Scanner s = new Scanner(System.in);

        int flightchoice = 0;

        while (flightchoice <= 0) {

            System.out.println("Enter the Flight choice :");

            if (s.hasNextInt()) {

                flightchoice = s.nextInt();

              if (flightchoice <= 0) {

                    System.out.println("Invalid Input... Please Enter Positive Number ");

                }

            } else {

                System.out.println("Invalid input . Please Enter Valid Number.");

            }

        }

        System.out.println("You selected Flight Number :" + flightchoice);

        s.close();

    }

}

/\* 4. Write a Java program that uses a for loop to print the first 10 numbers of the Fibonacci

sequence. \*/

public class unit\_2\_3 {

    public static void main(String[] args) {

        int a = 0, b = 1;

        System.out.println("First " + 10 + " Number of Fibonacci Sequence are :");

        for (int i = 0; i < 10; i++) {

            System.out.print(a + " ");

            int next = a + b;

            a = b;

            b = next;

        }

    }

}

// 5 . Create a Java program using a while loop to calculate the sum of integers from 1 to 100.

public class unit\_2\_4 {

    public static void main(String[] args) {

        int i = 1;

        int sum = 0;

        while (i <= 100) {

            sum += i;

            i++;

        }

        System.out.println("The Sum of Integer from 1 to 100 is :" + sum);

    }

}

/\* 6 . Implement a do-while loop that prompts the user to enter a number until they enter a

negative number. \*/

import java.util.Scanner;

public class unit\_2\_5 {

    public static void main(String[] args) {

        Scanner s = new Scanner(System.in);

        int number;

        System.out.println("Enter the Number : ");

        do {

            number = s.nextInt();

        } while (number >= 0);

        System.out.println("Negative Number Entered...");

        s.close();

    }

}

// 7 . Write a Java program that demonstrates the use of the continue statement in a loop.

public class unit\_2\_6 {

    public static void main(String[] args) {

        for (int i = 1; i < 10; i++) {

            if (i % 2 == 0) {

                continue;

            }

            System.out.println(i);

        }

    }

}

// 14 . Create a method that accepts an array and returns the maximum value using a for loop.

import java.util.Scanner;

public class unit\_2\_7 {

    public static void main(String[] args) {

        int maxvalue = 0;

        System.out.println("Enter the Size of Array :");

        int size = s.nextInt();

        int arr[] = new int[size];

        System.out.println("Enter the Elements of Array :");

        for (int i = 0; i < arr.length; i++) {

            arr[i] = s.nextInt();

            if (arr[i] > maxvalue) {

                maxvalue = arr[i];

            }

        }

        System.out.println("Maximum value is :" + maxvalue);

    }

}

// 15 . Write a Java program that finds the average of numbers stored in an integer array.

import java.util.Scanner;

public class unit\_2\_8 {

    public static void main(String[] args) {

        Scanner s = new Scanner(System.in);

        int sum = 0;

        double average;

        System.out.println("Enter the Size of Array :");

        int size = s.nextInt();

        int array[] = new int[size];

        System.out.println("Enter the Elements of Array :");

        for (int i = 0; i < size; i++) {

            array[i] = s.nextInt();

            sum += array[i];

        }

        average = (double) sum / size;

        System.out.println("The Average of Array Elements is :" + average);

        s.close();

    }

}

// 16 . Write a Java program that sums the elements of a 2D array.

import java.util.Scanner;

public class SimpleSum2DArray {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

          int sum = 0;

        // Read rows and columns

        System.out.print("Enter number of rows: ");

        int rows = scanner.nextInt();

        System.out.print("Enter number of columns: ");

        int cols = scanner.nextInt();

        int[][] array = new int[rows][cols];

        // Input elements of the 2D array

        System.out.println("Enter the elements:");

        for (int i = 0; i < rows; i++) {

            for (int j = 0; j < cols; j++) {

                array[i][j] = scanner.nextInt();

                sum += array[i][j];  // Directly add the element to sum

            }

        }

        System.out.println("Sum of elements: " + sum);

    }

}

// 19. Design a Java program that merges two sorted arrays into a single sorted array.

import java.util.Scanner;

public class unit\_2\_11 {

    public static void main(String[] args) {

        Scanner s = new Scanner(System.in);

        System.out.println("Enter the Size of First Array :");

        int size1 = s.nextInt();

        int array1[] = new int[size1];

        System.out.println("Enter the Elements of First Array :");

        for (int i = 0; i < size1; i++) {

            array1[i] = s.nextInt();

        }

        System.out.println("Enter the Size of Second Array :");

        int size2 = s.nextInt();

        int array2[] = new int[size2];

        System.out.println("Enter the Elements of Second Array :");

        for (int i = 0; i < size2; i++) {

            array2[i] = s.nextInt();

        }

        int[] mergedarray = new int[size1 + size2];

        for (int i = 0; i < size1; i++) {

            mergedarray[i] = array1[i];

        }

        for (int i = 0; i < size2; i++) {

            mergedarray[size1 + i] = array2[i];

        }

        System.out.println("Merged Array is :");

        for (int i = 0; i < mergedarray.length; i++) {

            System.out.print(mergedarray[i] + " ");

        }

        s.close();

    }

}

//20. Write a Java program to reverse the array

import java.util.Scanner;

public class unit\_2\_12 {

    public static void main(String[] args) {

        Scanner s = new Scanner(System.in);

        System.out.println("Enter the Size of Array :");

        int size = s.nextInt();

        int array[] = new int[size];

        System.out.println("Enter the Elements of Array :");

        for (int i = 0; i < size; i++) {

            array[i] = s.nextInt();

        }

        int start = 0;

        int end = size - 1;

        while (start < end) {

            int temp = array[start];

            array[start] = array[end];

            array[end] = temp;

            start++;

            end--;

        }

        for (int i = 0; i < size; i++) {

            System.out.print(array[i] + " ");

        }

        s.close();

    }

}

// 21. Find the Second largest element in Java

import java.util.Scanner;

public class unit\_2\_13 {

    public static void main(String[] args) {

        Scanner s = new Scanner(System.in);

        System.out.println("Enter the Size of Array :");

        int size = s.nextInt();

        int array[] = new int[size];

        System.out.println("Enter the Elements of Array :");

        for (int i = 0; i < size; i++) {

            array[i] = s.nextInt();

        }

        int largest = 0;

        int secondlargest = 0;

        for (int i = 0; i < array.length; i++) {

            if (array[i] > largest) {

                secondlargest = largest;

                largest = array[i];

            } else if (array[i] > secondlargest && array[i] != largest) {

                secondlargest = array[i];

            }

        }

        System.out.println("Second Larges element of array is :" + secondlargest);

        s.close();

    }

}

// 22 . Find the first even number in a list and breaks the loop when it finds.

public class unit\_2\_14 {

    public static void main(String[] args) {

        int i;

        for (i = 1; i < 100; i++) {

            if (i % 2 == 0) {

                break;

            }

        }

        System.out.println(i);

    }

}

// 23 . Prints all odd numbers from 1 to 20, using continue to skip even numbers.

public class unit\_2\_15 {

    public static void main(String[] args) {

        int i;

        for (i = 1; i < 20; i++) {

            if (i % 2 == 0) {

                continue;

            }

            System.out.println(i);

        }

    }

}

// 24. Prompts the user to enter numbers until they enter a negative number.

import java.util.Scanner;

public class unit\_2\_16 {

    public static void main(String[] args) {

        Scanner s = new Scanner(System.in);

        int number;

  System.out.println("Enter the Number : ");

        do {

            number = s.nextInt();

            if (number < 0) {

                System.out.println("You Entered Negative Number...");

                break;

            }

        } while (true);

        s.close();

    }

}

// 25. Prints a multiplication table but skips the multiplication by 5.

public class unit\_2\_17 {

    public static void main(String[] args) {

        int number = 5;

        System.out.println("Multiplication Table for " + number);

        for (int i = 1; i <= 10; i++) {

if (i == 5) { // Skip when multiplying by 5

Continue ;

}

            System.out.println(number + " X " + i + " = " + (number \* i));

        }

    }

}

// 26. Program counts from 1 to 10 but breaks when it reaches 6.

public class unit\_2\_18 {

    public static void main(String[] args) {

        for (int i = 1; i <= 10; i++) {

            if (i == 6) {

                break;

            }

            System.out.println(i);

        }

    }

}

// 27. Program prints numbers from 1 to 10 but skips the number 5.

public class unit\_2\_19 {

    public static void main(String[] args) {

        for (int i = 1; i <= 10; i++) {

            if (i == 5) {

                continue;

            }

            System.out.println(i);

        }

    }

}

/\* 28 . Develop a program that checks whether a given number is prime or not. Use a for loop

to test divisibility. If the number is found to be divisible by any number other than 1 and

itself, it is not prime.  \*/

import java.util.Scanner;

public class unit\_2\_20 {

    public static void main(String[] args) {

        Scanner s = new Scanner(System.in);

        System.out.println("Enter the Number :");

        int number = s.nextInt();

        boolean prime = true;

        if (number < 2) {

            prime = false;

        } else {

            for (int i = 2; i <= Math.sqrt(number); i++) {

                if (number % i == 0) {

                    prime = false;

                    break;

                }

            }

        }

        if (prime) {

            System.out.println(number + " is a prime Number.");

        } else {

            System.out.println(number + " is not a prime Number.");

        }

        s.close();

    }

}

/\* 29 . Create a program that reverses the digits of a given integer. Use a while loop to extract each digit and build the reversed number. \*/

import java.util.Scanner;

public class unit\_2\_21{

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        // Input: integer from the user

        System.out.print("Enter an integer: ");

        int number = scanner.nextInt();

        // Initialize variables for reversing

        int reversed = 0;

        // Use a while loop to reverse the digits

        while (number != 0) {

            int digit = number % 10; // Extract the last digit

            reversed = reversed \* 10 + digit; // Append the digit to reversed

            number /= 10; // Remove the last digit

        }

        // Output: the reversed number

        System.out.println("Reversed number: " + reversed);

    }

}

/\* 30. Write a program that prints the multiplication table for a given number. The user

should input the number and the range (e.g., up to 10 or 20). Use a for loop to generate

the table.  \*/

import java.util.Scanner;

public class unit\_2\_22 {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        // Input: number and range

        System.out.print("Enter the number for the multiplication table: ");

        int number = scanner.nextInt();

        System.out.print("Enter the range for the table: ");

        int range = scanner.nextInt();

        // Generate and print the multiplication table

        System.out.println("Multiplication Table for " + number + ":");

        for (int i = 1; i <= range; i++) {

            System.out.println(number + " X " + i + " = " + (number \* i));

        }

    }

}

/\*31. Write a program that counts the number of vowels and consonants in a given string.

Use a for loop to iterate through the string and keep track of the counts.  \*/

import java.util.Scanner;

public class unit\_2\_23 {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        // Input: string from the user

        System.out.print("Enter a string: ");

        String input = scanner.nextLine();

        // Initialize counters

        int vowelCount = 0;

        int consonantCount = 0;

        // Convert the string to lowercase for easier comparison

        input = input.toLowerCase();

        // Iterate through each character in the string

        for (int i = 0; i < input.length(); i++) {

            char ch = input.charAt(i);

            // Check if the character is a vowel

            if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u') {

                vowelCount++;

            }

            // Check if the character is a consonant (alphabetic but not a vowel)

            else if (ch >= 'a' && ch <= 'z') {

                consonantCount++;

            }

        }

        // Output: number of vowels and consonants

        System.out.println("Number of vowels: " + vowelCount);

        System.out.println("Number of consonants: " + consonantCount);

    }

}

**32 Print the Pattern as per given etc**

**1 1 1 1 1**

**1 1 1 1**

**1 1 1**

**1 1**

**1**

/\*Print the Pattern as per given etc

1 1 1 1 1

1 1 1 1

1 1 1

1 1

1

 \*/

public class Pattern {

    public static void main(String[] args) {

        int rows = 5; // Number of rows for the pattern

        // Outer loop for rows

        for (int i = rows; i >= 1; i--) {

            // Inner loop to print "1"

            for (int j = 1; j <= i; j++) {

                System.out.print("1 ");

            }

            // Move to the next line after each row

            System.out.println();

        }

    }

}

/\*

\* \* \* \*

\* \*

\* \*

\* \* \* \*

\*/

import java.util.Scanner;

public class Rectanglepattern {

    public static void main(String[] args) {

        Scanner s = new Scanner(System.in);

        System.out.println("Enter the number for rows:");

        int n = s.nextInt();

        System.out.println("Enter the number of columns:");

        int m = s.nextInt();

        for (int i = 1; i <= n; i++) {

            for (int j = 1; j <= m; j++) {

                if (i == 1 || i == n || j == 1 || j == m) {

                    System.out.print("\* ");

                } else {

                    System.out.print("  ");

                }

            }

            System.out.println();

        }

    }

}

**33. As part of a feedback collection system, you want to gather customer ratings for a product. Design a program that prompts customers to rate the product from 1 to 5. Use a labeled while loop to continue collecting ratings until a customer enters 0. After collecting all ratings, compute and display the average rating and the number of ratings received.**

/\*As part of a feedback collection system, you want to gather customer ratings

for a product. Design a program that prompts customers to rate the product from

1 to 5. Use a labeled while loop to continue collecting ratings until a customer

enters 0. After collecting all ratings, compute and display the average rating

And the number of ratings received.

 \*/

import java.util.Scanner;

public class Feedback {

    public static void main(String[] args) {

        Scanner s=new Scanner(System.in);

        int totalRating=0;

        int count =0;

        System.out.println("Enter the Rating : ");

ratingsLoop:

        while (true) {

            int rating=s.nextInt();

            if(rating==0){

              break ratingsLoop;            }

            if(rating >= 1 && rating <= 5){

                totalRating+=rating;

                count++;

            }

        }

        if(count>=1){

            double averagerating=(double)totalRating/count;

            System.out.println("Average Rating is : "+averagerating);

            System.out.println("Total Number of Rating Received : "+count);

        }

        else{

            System.out.println("No Rating Received...");

        }

    }

}

1. **You are tasked with developing a program that tracks a user's monthly expenses. The program should repeatedly ask the user to input their expenses for different categories (like food, transportation, etc.) until they type "done". After the user is finished, display the total expenses for the month.**

/\*34.You are tasked with developing a program that tracks a user's monthly expenses.

The program should repeatedly ask the user to input their expenses for different categories

(like food, transportation, etc.) until they type "done". After the user is finished,

display the total expenses for the month. \*/

import java.util.\*;

public class MonthExpenses {

    public static void main(String[] args) {

        Scanner s=new Scanner(System.in);

    int totalExpense=0;

    while (true) {

        System.out.println("Enter the Expense Category or Type 'done' to finish :" );

        String  Category =s.nextLine();

        if(Category.equalsIgnoreCase("done")){

            break;

        }

        System.out.println("Enter the Amount for "+Category+" :");

        double Amount=s.nextInt();

        totalExpense+=Amount;

        s.nextLine();

    }

        System.out.println("Total Expenses For Month is : $ "+totalExpense);

    }

}

**35. Develop a password validation system that prompts users to create a password. The program should check if the password meets certain criteria (length, special characters, etc.). If it doesn't meet the criteria, it should continue prompting the user until a valid password is entered.**

/\* Develop a password validation system that prompts users to create a password.

The program should check if the password meets certain criteria (length, special characters, etc.). If it doesn't meet the criteria, it should continue prompting the user until a valid password is entered. \*/

import java.util.Scanner;

public class Password {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        String password;

        // Loop until a valid password is entered

        while (true) {

            System.out.print("Enter your password: ");

            password = scanner.nextLine();

            // Check if the password is valid

            if (password.length() >= 8

                && password.matches(".\*[A-Z].\*")

                && password.matches(".\*[a-z].\*")

                && password.matches(".\*[0-9].\*")

                && password.matches(".\*[@#$%^&+=].\*")) {

                System.out.println("Password is valid!\nPassword Successfully Created..");

                break;

            } else {

                System.out.println("Invalid password. Make sure it is at least 8 characters long, contains uppercase, lowercase, a digit, and a special character.");

            }

        }

        scanner.close();

    }

}

**36 .Create a fitness app that allows users to log their daily steps. The user should be prompted to enter their steps for each day of the week. Use a loop to collect this data, and at the end of the week, calculate and display the total steps taken and the average steps per day.**

/\*Create a fitness app that allows users to log their daily steps. The user should be prompted to enter their steps for each day of the week. Use a loop to collect this data, and at the end of the week, calculate and display the total steps taken and the average steps per day. \*/

import java.util.Scanner;

public class FitnessApp {

    public static void main(String[] args) {

        Scanner s=new Scanner(System.in); String[]daysOfTheWeek={"Monday","Tuesday","Wednesday","Thursday","Friday","Saturday","Sunday"};

        int steps[]=new int[7];

    int totalSteps=0;

    for(int i=0;i<daysOfTheWeek.length;i++){

    System.out.print("Enter the Steps For "+daysOfTheWeek[i]+" : ");

    steps[i]=s.nextInt();

    totalSteps+=steps[i];

    }

    double average=(double)totalSteps/daysOfTheWeek.length;

    System.out.println("Total Steps in Week : "+totalSteps);

    System.out.println("Average Steps in a Day is : "+average);

    }

}

**37. Develop a temperature conversion tool that allows users to convert temperatures between Celsius and Fahrenheit. Use a loop to continue asking for temperature values until the user chooses to exit. After each conversion, display the result and prompt the user again.**

/\*Develop a temperature conversion tool that allows users to convert temperatures between

Celsius and Fahrenheit. Use a loop to continue asking for temperature values until the user

chooses to exit. After each conversion, display the result and prompt the user again.  \*/

import java.util.Scanner;

public class temperatureConverter {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        while (true) {

            System.out.println("Temperature Conversion Tool");

            System.out.println("1. Convert Celsius to Fahrenheit");

            System.out.println("2. Convert Fahrenheit to Celsius");

            System.out.println("3. Exit");

            System.out.print("Choose an option (1, 2, or 3): ");

            int choice = scanner.nextInt();

            switch (choice) {

  case 1:

                    // Celsius to Fahrenheit

                    System.out.print("Enter temperature in Celsius: ");

                    double celsius = scanner.nextDouble();

                    double fahrenheit = (celsius \* 9 / 5) + 32;

                    System.out.println(celsius + "°C = " + fahrenheit + "°F");

                    break;

                case 2:

                    // Fahrenheit to Celsius

                    System.out.print("Enter temperature in Fahrenheit: ");

                    fahrenheit = scanner.nextDouble();

                    celsius = (fahrenheit - 32) \* 5 / 9;

                    System.out.println(fahrenheit + "°F = " + celsius + "°C");

                    break;

                case 3:

                    // Exit the program

                    System.out.println("Exiting the program.");

                    scanner.close();

                    return; // End the program

                default:

                    System.out.println("Invalid option. Please choose 1, 2, or 3.");

                    break;

            }

        }

    }

}

1. **Implement a simple banking system where users can deposit and withdraw money. Use a loop to allow the user to perform transactions until they choose to exit. After exiting, display the final account balance and transaction history.**

/\*Implement a simple banking system where users can deposit and withdraw money.

Use a loop to allow the user to perform transactions until they choose to exit.

After exiting, display the final account balance and transaction history. \*/

import java.util.Scanner;

public class ab {

     public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        double balance = 0.0;

        String transactionHistory = "";

        while (true) {

            System.out.println("\nSimple Banking System");

            System.out.println("1. Deposit");

            System.out.println("2. Withdraw");

            System.out.println("3. Check Balance");

            System.out.println("4. Exit");

            System.out.print("Choose an option (1-4): ");

            int choice = scanner.nextInt();

            if (choice == 1) {

                System.out.print("Enter amount to deposit: ");

                double deposit = scanner.nextDouble();

                    balance += deposit;

                    transactionHistory += "Deposited: $" + deposit + "\n";

                    System.out.println("Deposited: $" + deposit);

            } else if (choice == 2) {

                System.out.print("Enter amount to withdraw: ");

                double withdraw = scanner.nextDouble();

                if (withdraw <= balance) {

                    balance -= withdraw;

                    transactionHistory += "Withdrew: $" + withdraw + "\n";

                    System.out.println("Withdrew: $" + withdraw);

                } else {

                    System.out.println("Insufficient balance!");

                }

            } else if (choice == 3) {

                System.out.println("Current Balance: $" + balance);

            } else if (choice == 4) {

                System.out.println("\nFinal Balance: $" + balance);

                System.out.println("Transaction History:\n" + transactionHistory);

                System.out.println("Thank you for using the Simple Banking System. Goodbye!");

                break;

            } else {

                System.out.println("Invalid option. Please try again.");

            }

        }

        scanner.close();

    }

}

1. **Create a program that allows a teacher to input grades for students in a class. The program should continue to prompt for grades until the teacher enters -1 to stop. After all grades have been entered, calculate and display the average grade, the highest grade, and the number of students who passed (e.g., scored above a certain threshold). share code**

/\* Create a program that allows a teacher to input grades for students in a class.

The program should continue to prompt for grades until the teacher enters -1 to stop.

After all grades have been entered, calculate and display the average grade, the highest grade,

and the number of students who passed (e.g., scored above a certain threshold). \*/

import java.util.\*;

public class greadTracker {

    public static void main(String[] args) {

        Scanner s=new Scanner(System.in);

        int sum=0,highestGrade=0,passCount=0,totalGrade=0;

        int passingThreshold=40;

        int choice;

        do {

            System.out.println("Grade Tracking System...");

            System.out.println("1 : To Enter the Grade of Students : ");

            System.out.println("2 : To Show Result and Exit..");

            System.out.println("\nEnter the Choice : ");

            choice=s.nextInt();

            switch (choice) {

                case 1:

                    System.out.println("Enter the Grade");

                    int grade=s.nextInt();

                    if(grade==-1){

                        choice=2;

                        break;

                    }

                    if(grade<0||grade>100){

                        System.out.println("Invalid Grade");

                        break;

                    }

                    else{

                    sum +=grade;

                    totalGrade++;

                if(grade > highestGrade){

                    highestGrade = grade;

                }

                if(grade >= passingThreshold){

                    passCount++;

                }

            }

                    break;

                case 2:

                    if(totalGrade>0){

                    double average =(double)sum/totalGrade;

                    System.out.println("\nGrade Summary...");

                    System.out.println("Highest Grade is : "+highestGrade);

                    System.out.println("Average is : "+average);

                    System.out.println("Number of Student Pass : "+passCount);

                    }

                    else{

                        System.out.println("NO Grade has Entered.");

                    }

                    break;

                default:

                    System.out.println("Invalid Option.");

            break;

                }

        }while (choice!=2);

    }

}

**40. Design a shopping cart application that allows users to add items to their cart. The program should ask the user for item names and prices in a loop until the user types**

/\*Design a shopping cart application that allows users to add items to their cart.

The program should ask the user for item names and prices in a loop until the user types \*/

import java.util.\*;

public class Shop ingChart {

    public static void main(String[] args) {

        Scanner s=new Scanner(System.in);

        String item ="";

        double totalAmount=0,itemPrice=0;

        String itemName;

        int choice;

        while (true) {

            try {

                System.out.println("\nShoping APP..");

                System.out.println("1 : ADD To Chart");

                System.out.println("2 : Check Out");

                System.out.println("3 : Exit Without Check Out...");

                System.out.println("\nEnter the Choice : ");

                choice=s.nextInt();

                s.nextLine();

            switch (choice) {

                case 1:

                    System.out.println("Enter the Item Name : ");

                    itemName=s.nextLine();

                    System.out.println("Enter the Amount for "+itemName+" : ");

                    itemPrice=s.nextDouble();

                    item+=itemName+"($"+itemPrice+")\n";

                    totalAmount+=itemPrice;

                    break;

                case 2:

                    System.out.println("Item Purchased :\n"+item);

                    System.out.println("Total Amount Due :"+totalAmount);

                    return;

                case 3:

                    System.out.println("Exiting Without Check Out...");

                    return;

                    default:

                    System.out.println("Invalid Choice...");

                    break;

            }

        } catch (Exception e) {

            System.out.println("Invalid Input! Please Try Again");

            s.nextLine();

        }

        }

    }

}

1. **Write a program that calculates the total sales and commission for a group of salespeople. Prompt the user to enter sales figures for each salesperson in a loop. The loop should continue until a negative number is entered, indicating the end of input. Calculate and display the total sales and the average sales per salesperson.**

/\*Write a program that calculates the total sales and commission for a group of salespeople.

Prompt the user to enter sales figures for each salesperson in a loop. The loop should

continue until a negative number is entered, indicating the end of input.

 Calculate and display the total sales and the average sales per salesperson. \*/

import java.util.Scanner;

public class SalesCalculator {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        double totalSales = 0.0;

        int salespersonCount = 0;

        while (true) {

            // Ask for sales input

    System.out.print("Enter sales for salesperson   (or enter a negative number to stop): ");

            double sales = scanner.nextDouble();

            // If the entered sales number is negative, break the loop

            if (sales < 0) {

                break;

            }

            // Add sales to total and count the salesperson

            totalSales += sales;

            salespersonCount++;

        }

        // Display total sales and average sales

        if (salespersonCount > 0) {

            double averageSales = totalSales / salespersonCount;

            System.out.println("\nTotal Sales: $" + totalSales);

            System.out.println("Average Sales per Salesperson: $" + averageSales);

        } else {

            System.out.println("No sales data entered.");

        }

        scanner.close();

    }

}

**42. Write a Java program to reverse a String.**

import java.util.Scanner;

public class QB\_42 {

    public static void main(String[] args) {

        // Create a scanner to read input

        Scanner scanner = new Scanner(System.in);

        // Ask for the user input string

        System.out.print("Enter a string: ");

        String input = scanner.nextLine();

        // Reverse the string using StringBuilder

        String reversed = new StringBuilder(input).reverse().toString();

        // Display the reversed string

        System.out.println("Reversed String: " + reversed);

        // Close the scanner

        scanner.close();

    }

}

**43. How would you check if a String is a palindrome in Java?**

//How would you check if a String is a palindrome in Java

import java.util.Scanner;

public class PalindromeExample {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter a string: ");

        String input = scanner.nextLine();

        // Check if the string is a palindrome

        if (input.equalsIgnoreCase(new StringBuilder(input).reverse().toString())) {

            System.out.println("The string is a palindrome.");

        } else {

            System.out.println("The string is not a palindrome.");

        }

        scanner.close();

    }

}

**44. How would you identify and count the occurrences of each character in a String?**

import java.util.Scanner;

public class QB\_44 {

    public static void main(String[] args) {

        // Create a scanner to take input from the user

        Scanner scanner = new Scanner(System.in);

        // Ask the user for the string

        System.out.print("Enter a string: ");

        String input = scanner.nextLine();

        // Loop through each character in the string

        for (int i = 0; i < input.length(); i++) {

            char currentChar = input.charAt(i);

            int count = 0;

            // Count occurrences of the current character

            for (int j = 0; j < input.length(); j++) {

                if (input.charAt(j) == currentChar) {

                    count++;

                }

            }

            // Print the character and its count only once

            if (input.indexOf(currentChar) == i) {

                System.out.println(currentChar + ": " + count);

            }

        }

        // Close the scanner

        scanner.close();

    }

}

1. **Write a Java program to reverse a given String without using the built-in reverse method.**

**Example: Input: "Hello" Output: "olleH"**

import java.util.Scanner;

public class QB\_45 {

    public static void main(String[] args) {

        // Scanner to take input from the user

        Scanner scanner = new Scanner(System.in);

        // Ask for the string input

        System.out.print("Enter a string: ");

        String input = scanner.nextLine();

        // Variable to store the reversed string

        String reversed = "";

        // Loop through the string from end to start

        for (int i = input.length() - 1; i >= 0; i--) {

            reversed += input.charAt(i);  // Append each character to the reversed string

        }

        // Display the reversed string

        System.out.println("Reversed String: " + reversed);

        // Close the scanner

        scanner.close();

    }

}

1. **Create a method that checks if a given String is a palindrome (reads the same forwards and backwards).**

**Input: "racecar" Output: true**

import java.util.Scanner;

public class QB\_45 {

    public static void main(String[] args) {

        // Scanner to take input from the user

        Scanner scanner = new Scanner(System.in);

        // Ask for the string input

        System.out.print("Enter a string: ");

        String input = scanner.nextLine();

        // Check if the string is a palindrome

        if (new StringBuilder(input).reverse().toString().equals(input)) {

            System.out.println("The string is a palindrome.");

        } else {

            System.out.println("The string is not a palindrome.");

        }

        // Close the scanner

        scanner.close();

    }

}

1. **Write a program that counts the number of vowels and consonants in a given String.**

**Input: "Hello World" Output: Vowels: 3, Consonants: 7**

public class QB {

    public static void main(String[] args) {

        // Scanner to take input from the user

        Scanner scanner = new Scanner(System.in);

        // Ask for the string input

        System.out.print("Enter a string: ");

        String input = scanner.nextLine().toLowerCase();

        int vowels = 0, consonants = 0;

        // Loop through each character in the string

        for (int i = 0; i < input.length(); i++) {

            char ch = input.charAt(i);

            // Check if the character is a vowel or consonant

            if (ch >= 'a' && ch <= 'z') {

                if ("aeiou".indexOf(ch) != -1) {

                    vowels++;

                } else {

                    consonants++;

                }

            }

        }

        // Output the counts of vowels and consonants

        System.out.println("Vowels: " + vowels + ", Consonants: " + consonants);

        // Close the scanner

        scanner.close();

    }

}

**48 Implement a method that capitalizes the first letter of each word in a given String.**

**Input: "hello world" Output: "Hello World"**

import java.util.Scanner;

public class QB {

    public static void main(String[] args) {

        // Scanner to take input from the user

        Scanner scanner = new Scanner(System.in);

        // Ask for the string input

        System.out.print("Enter a string: ");

        String input = scanner.nextLine();

        // Capitalize the first letter of each word

        String result = capitalizeFirstLetter(input);

        // Output the result

        System.out.println("Capitalized String: " + result);

        // Close the scanner

        scanner.close();

    }

    // Method to capitalize the first letter of each word

    public static String capitalizeFirstLetter(String str) {

        String[] words = str.split(" ");

        StringBuilder capitalizedString = new StringBuilder();

        for (String word : words) {

            if (!word.isEmpty()) {

             capitalizedString.append(Character.toUpperCase(word.charAt()

                        .append(word.substring(1))

                        .append(" ");

            }

        }

        return capitalizedString.toString().trim();

    }

}

1. **Implement a method that checks if two Strings are anagrams of each other (contain the same characters in a different order).**

**Input: "listen", "silent" Output: true**

import java.util.Arrays;

public class QB {

    public static void main(String[] args) {

        // Test strings

        String str1 = "listen";

        String str2 = "silent";

        // Check and print if the strings are anagrams

        System.out.println(areAnagrams(str1, str2));

    }

    // Method to check if two strings are anagrams

    public static boolean areAnagrams(String str1, String str2) {

        return str1.length() == str2.length()

                && Arrays.equals(

                        str1.chars().sorted().toArray(),

                        str2.chars().sorted().toArray());

    }

}

**50. Write a program to remove duplicate characters from a String while maintaining the original order of characters. Input: "programming" Output: "progamin"**

public class QB {

    public static void main(String[] args) {

        String input = "omsairam";

        System.out.println(removeDuplicates(input));

    }

    // Method to remove duplicate characters

    public static String removeDuplicates(String str) {

        StringBuilder result = new StringBuilder();

        for (char ch : str.toCharArray()) {

            if (result.indexOf(String.valueOf(ch)) == -1) {

                result.append(ch);

            }

        }

        return result.toString();

    }

}